Statistical design of pools using optimal coverage and minimal collision

— Source link ↗

Katja Remlinger, Jacqueline M. Hughes-Oliver, S. Stanley Young, Raymond L. H. Lam

Related papers:

- The Detection of Defective Members of Large Populations

- Regression analysis of group testing samples.

- Regression models for disease prevalence with diagnostic tests on pools of serum samples.

- Comparison of group testing algorithms for case identification in the presence of test error.

- Group testing regression models with fixed and random effects.

Statistical Design of Pools Using Optimal Coverage and Minimal Collision -- Institutue of Statistics Mimeo Series 2549

Katja S. Remlinger[1], Jacqueline M. Hughes-Oliver[1],

S. Stanley Young[2], Raymond L. H. Lam[3]

[1]Department of Statistics, North Carolina State University, Raleigh, NC 27695-8203

[2]National Institute of Statistical Sciences, Research Triangle Park, NC 27709-4006

[3]Data Exploration Sciences, GlaxoSmithKline, King of Prussia, PA 19406

Abstract

Discovery of a new drug involves screening large chemical libraries to identify active compounds. Screening efficiency can be improved by testing compounds in pools. We consider two criteria to design pools: optimal coverage of the chemical space and minimal collision between compounds. Five pooling designs are applied to a public data set. We evaluate each method by determining how well the design criteria are met. One design uniformly dominates all other designs, but all criteria-designed pools outperform randomly created pools. Furthermore, we discuss blocking and synergism between compounds as other effects that must be investigated when performing pooling experiments.

KEY WORDS: Blocking; Drug discovery; High throughput screening; Pooling experiment; Synergism; Uniform cell coverage design.

---

1 INTRODUCTION

The drug discovery process starts with screening chemical libraries (collections of chemical compounds) to identify new lead compounds. Compounds are tested for potency with respect to one or several biological targets in an automated process known as high throughput screening (HTS). This process involves screening thousands to tens of thousands of chemical compounds per week. However, today's chemical libraries are extremely large, on the order of hundreds of thousands or millions of compounds, so that even the high throughput screening process can take months to test the entire library. This, and the fact that only a small fraction of the compounds in a library are actually active, leads to the conclusion that this approach is not very cost or time efficient. New methods for improving screening efficiency are needed. Xie, Tatsuoka, Sacks, and Young (2001) state that, in situations when the rate of active individuals is very small but the number of individuals to be tested is very large, testing individuals in groups is an effective alternative to testing them one by one. This scenario describes exactly the situation we are facing when screening large collections of chemical compounds. Therefore, one alternative approach to assaying compounds individually is to test them in pools, for example put 10 compounds together for testing. Creating the pools can be done either in a random fashion, or by applying certain design criteria.

The chemical properties of every compound in the library can be described by a variety of molecular descriptors. McFarland and Gans (1986) note that two compounds with fairly close values of all critical descriptors are more likely to have similar biological activity. Additionally, scientists believe that there is a correlation between the activity of a compound and its chemical structure (structure-activity relationship, SAR). In light of this, it is especially important to choose descriptors that are informative about the activity of a compound with respect to the biological targets under consideration. In this way, the chem-

1

ical descriptors of a compound can be used to guide the search for lead compounds when assaying chemical libraries. Descriptors that are commonly used are atom pairs (Carhart, Smith, and Ventkataraghavan 1985), topological torsions (Nilakantan, Bauman, Dixon, and Ventkataraghavan 1987), and the continuous BCUT numbers. BCUTs are described by Pearlman and Smith (1998, 1999) and originally derived by Burden (1989).

In this paper, we propose two design criteria for creating pools. The first criterion tries to cover the chemical space as uniformly as possible when choosing compounds to be assigned to pools. The second criterion aims for minimizing collision among compounds that form a pool, i.e., minimize the similarity of compounds within a pool. A discussion will follow on why we think it is important to fulfill these two design criteria, and what we can gain by doing so. Lam, Welch, and Young (2002), hereafter LWY, propose a method for uniformly covering low dimensional subspaces. They use an exchange algorithm to optimize the criterion for uniformity of coverage. We use this algorithm as a tool to select the compounds we want to assign to each pool. We propose and compare four different pooling designs that fulfill at least one of the design criteria. Performance of the four pooling designs is illustrated using a data set from the National Cancer Institute (NCI). Table 1 gives an overview of the four proposed pooling designs, indicating which design criteria are fulfilled by each design. A fifth pooling design is included in the table. This design, Total Random (TR), does not satisfy any of the two design criteria and it is used as a benchmark pooling design.

[Table 1 about here.]

Section 2 contains an introduction to pooling experiments. Section 3 explains the two proposed design criteria, namely, optimal coverage and minimal collision. We also discuss the importance and benefits of these criteria. Section 4 describes the NCI data, including

the chemical descriptors used and why they were chosen. In Section 5, a data-driven binning approach is described. This method is used to create suitable descriptors that allow assessment of how well the design criteria are fulfilled. This section also describes the SpaceFill algorithm proposed by LWY and how we use it to choose compounds that fulfill the design criteria. In Section 6, we provide a detailed discussion of the four pooling designs. We illustrate their realization on the NCI data set, and compare how well they fulfill the design criteria. Finally, in Section 7 we close with a summary and discussion of future work.

2 POOLING EXPERIMENTS

Pooling experiments, also known under the name of group testing, were first introduced by Dorfman in 1943. He suggested testing men's blood samples for syphilis by pooling them into groups rather than testing each sample individually. If a pool tests active, the individuals will be retested so that the active individual sample(s) can be determined. Dorfman (1943) illustrated the cost and time savings that could be achieved by applying this procedure. Phatarfod and Sudbury (1994) discuss pooling in the context of HIV testing when pooled tests are flawed. Many other authors (for example, Tu, Litvak, and Pagano 1995; Behets et al. 1990; Hughes-Oliver and Rosenberger 2000) also discuss pooling as a means of screening for other diseases, even simultaneously screening for several diseases, in the presence of flawed tests. Findings clearly support superiority of pooling experiments over one-at-a-time testing.

Today, pooling methods are used to identify active individuals in many other different applications. Langfeldt, Hughes-Oliver, Ghosh and Young (1997), Hann et al. (1999), Xie et al. (2001), Zhu, Hughes-Oliver, and Young (2001), and Yi (2002) note that the pooling methodology is commonly used in the drug discovery process, and they demonstrate its use on real, not simulated, pooled data from a large pharmaceutical company. Pooling is used

3

to screen large chemical libraries, and is considered to be a very efficient high throughput screening technique.

Pooling experiments for chemical compounds proceed in a highly automated fashion. First, the solid components need to be liquified by mixing them with a solvent. Robotic systems place tiny amounts of the solutions into different wells arranged on a plate. In one-way pooling, either all samples along one column are combined to form a pool, or all samples along one row. This is done for all the columns or rows respectively. If a plate contains, for example, n columns, a total of n pools are formed. In our experiments, we typically combine 10 samples into one pool. Two-way pooling combines the samples along the m rows and the samples along the n columns to create a total of $m+n$ pools. Two-way pooling can be used to facilitate decoding of actives and to decrease the chance of false inactive pools (Langfeldt et al. 1997; Phatarfod and Sudbury 1994; Xie et al. 2001); false inactive pools are more fully discussed below. We only consider one-way pooling in this paper. Figure 1 shows a typical one-way pooling design where the samples of each column are combined into one pool.

[Figure 1 about here.]

The chemical compounds that we group into one pool are all expected to differ in their activities with respect to a given biological target; most compounds are either completely inactive or only weakly active. Dorfman (1943) and many others define a pool to be active if at least one of the compounds in the pool is active, and inactive if none of the compounds in the pool are active. When pooling chemical compounds, however, these simplifying assumptions are not necessarily valid. It can occur that some pools are tested as inactive although they contain active compounds, because one compound in a pool is blocking the activity of another compound in the same pool. Langfeldt et al. (1997) call compounds that prevent

4

the detection of active compounds blockers. In other words, blockers lead to false inactive pools. There are a number of possible blocking mechanisms.

Compounds are mixed with a solvent when placing them into a pool. This liquid state allows easier handling of the compounds. But the liquid state, which is necessary for the assay, allows two compounds to react and join together and in this combined state they may no longer be active. Suppose, for example, the goal is to identify compounds with the ability to bind to a particular receptor. It can happen that an individually tested compound has this binding ability, but when placed into a solution with other compounds the active compound reacts with a blocker compound and can no longer fit the binding site. It is also possible that the blocker alters the primary binding site, so that the active compound can no longer bind to it. Blocking is illustrated in Pool 8 of Figure 1. Column 8 contains an active compound, as well as a blocker compound. As a result, the pooled compounds do not show any activity. Blocking has also been researched in other applications of pooling methods, such as blood testing experiments (Phatarfod and Sudbury 1994).

Another effect that can occur when placing compounds into pools is synergism. Two chemically dissimilar compounds are synergistic to one another if, when combined, they show activity, but individual testing reveals no activity or only weak activity. Again, synergism can occur because two compounds combined in one pool can react with each other and create a new compound for binding at the site of interest. One compound might bind at a secondary binding site and alter the shape of the primary binding site. Evidence of synergism and researchers' interest in it is given in various forms. While trying to understand the mechanistic actions of p68 RNA helicase, Rossow and Janknecht (2003) discover synergistic relationships between p68 transcriptional coactivators p300 and CERB-binding proteins. Tipping et al. (2002) discuss synergistic relationships between imatinib, a powerful inhibitor of the tyrosine kinase activity of the oncoprotein responsible for chronic myeloid leukemia,

5

and three other compounds. Furthermore, the company ActivBiotics (2003) is currently developing rifalazil, a best-in-class antibiotic that targets and inhibits the transcriptional machinery of bacteria. They report that rifalazil acts synergistically with other classes of antibiotics. Figure 1 illustrates the concept of synergism. None of the individual compounds in column 1 show activity, yet the pooled compounds are active; that is, synergism occurs.

We think it is very important to take these two effects, blocking and synergism, into consideration when performing pooling experiments or statistical analysis of pooled data. Langfeldt et al. (1997), Zhu (2000), and Xie et al. (2001) all assume the presence of blockers in their studies of optimal pooling strategies and cost efficiency. Xie et al. (2001) also consider the effects of synergistic compounds in their work, and they say that at the early stage of drug discovery, compounds acting synergistically are usually considered a source of contamination. We, however, view synergism as a critical path to identifying "combination therapies." Yi (2002) considers blocking and synergism when modeling pooled and individual activity of chemical compounds.

Testing compounds in pools has several advantages. As already attributed to Dorfman (1943), pooling can bring very large benefits in terms of cost and time efficiency, but at the cost of complexity. The chemical libraries of pharmaceutical companies are of the order of millions of compounds. It would take months to screen each compound individually against a particular biological target. Testing them in groups of 10 significantly decreases the time needed to screen the entire library. Very often, only a small fraction of the compounds in a chemical collection are active with respect to the biological target assayed. Consequently, when testing the compounds in groups, only a small number of pools will contain an active compound, which in turn means that only relatively few pools will test active and need to be decoded. When a pharmaceutical company tries to find new lead compounds, it typically chooses only the active pools, and individually retests all compounds in the active

6

pools. One can see that this procedure considerably reduces the number of tests performed, especially when the proportion of active compounds is very small in comparison to non-active compounds. Since a certain cost can be assigned to each test, this approach does not only bring a reduction in time spent, but also a reduction of the cost. Chemical libraries will continue to increase in size in the years to come, and therefore we think it is important to investigate new ways of screening large chemical libraries.

A number of diseases such as HIV, hepatitis, and cancer are treated with multiple drugs in what is called "combination therapy." Several different drugs are simultaneously given to the patient. Although each drug is typically aimed towards a different biological target, we think nevertheless it is important to consider the blocking and synergistic properties of different compounds. As mentioned earlier, pooling experiments provide the opportunity to learn more about the blocker and synergism probabilities among compounds at an early stage of the drug discovery process.

To make pooling designs as efficient as possible, certain design criteria can be applied to assign individual compounds to pools. In the next section, we discuss two such criteria and explain why we think they are important.

3   POOLING DESIGN CRITERIA

Typically, when chemical libraries are screened for activity against a particular biological target, only a subset of all available compounds is selected and the activity of the compounds in this set is determined. Since only a subset of the collection of compounds is screened, it is desirable to select compounds that come from different regions of the chemical space. Cummins, Andrews, Bentley, and Cory (1996) and Menard, Mason, Morize, and Bauerschmidt (1998) provide cell-based methods for choosing screening sets of diverse compounds from large databases. In pooling experiments, a similar approach can be used; divide the space

7

into cells and select a compound from each occupied cell. In particular, we select a subset of the entire chemical library and assign these selected compounds into pools. The pools are then tested for activity. Again, it is important that the chemical space is covered as completely as possible by the subset of compounds that will be screened, which are also the compounds that will be assigned into pools.

Certain chemical descriptors are calculated for all the compounds in the library. Using the chemical descriptors as covariates, the selection of compounds that cover the chemical space is guided by the chemical descriptors. Furthermore, we do not want to lose any information by pooling the compounds. This can be assured by avoiding assignment of compounds that live in the same area of the chemical space into the same pool; two compounds with similar features should be in different pools so they give replication and do not inadvertently increase the concentration of the tested compound (we want each compound/chemical feature tested at the same concentration). The above considerations lead to our two proposed design criteria: Optimal Coverage and Minimal Collision.

3.1 Optimal Coverage

When chemical compounds are selected in such a way that they represent the chemical space as completely as possible, we say they cover the chemical space, or the chemical space is covered. Coverage of the chemical space is important because potent compounds can appear anywhere in the chemical space, not only in regions of high compound density. Very often, chemical synthesis is based on previous efforts to detect active compounds. This results in chemical libraries that have areas that are dense in compounds because an active was found in a particular area and similar compounds were made. A new query typically involves a different biological target, and might require compounds from a different area in the chemical space. When choosing compounds at random, compounds from highly dense subspaces are

more likely to be selected and a randomly selected data set will over-represent these areas. However, if we select members of the data set in such a way that compounds from all regions of the chemical space are uniformly represented, this data set will assure that active compounds in sparse subspaces have a better chance of being detected. Implementation of this criterion is discussed in Section 5.2.

One problem that occurs is that in high-dimensional spaces it is essentially impossible to cover the complete space by only selecting thousands of design points. Therefore, we will concentrate only on low-dimensional projections of the space, following the suggestion of LWY. This will be discussed in more detail in Sections 4 and 5.1.

3.2   Minimal Collision

Two compounds are said to be colliding when they have one or more very similar descriptor values. Imagine splitting the range of a descriptor into a certain number of cells. We then could say that two compounds are colliding if they live in the same cell. When creating pools, we want to minimize the number of collisions within a pool by selecting compounds that are as diverse as possible. Implementation of this criterion is discussed in Section 5.2.

There are several reasons why we think it is important to fulfill this design criterion. First, when creating the pools, we want to avoid additive effects. Compounds that are weakly active might appear as highly active when they are pooled together, because their features are effectively tested at a much higher concentration. These effects might be misleading in the process of identifying new lead compounds.

Another reason for trying to minimize collisions within the pools is to avoid ambiguous decoding. A mathematical decoding model can be used to point to specific compounds in the pool and obviate testing of each compound in the pool. If several compounds within an active pool share the same chemical features, it will be hard to identify the specific

9

compound that induces the activity in the pool without retesting each compound in the pool.

4   NCI DATA AND CHEMICAL DESCRIPTORS

We illustrate the realization of our proposed design criteria on the DTP AIDS Antiviral Screen database from the National Cancer Institute (NCI). The database contains screening results for evidence of anti-HIV activity, and the chemical structure of 43,905 compounds. More information on this public data set can be obtained from `http://dtp.nci.nih.gov/docs/aids/aids_data.html`. We use an older version of this database (May 1999), which contained originally 32,110 compounds. Only 29,749 of these compounds were used by us as the descriptors could not be calculated for all members of the database. We will provide this data set to those that are interested.

As mentioned earlier, an important decision is choice of appropriate molecular descriptors that describe the chemical properties of each compound in the chemical library. We need descriptors that capture structural features of a compound that are relevant to activity. BCUT numbers seem the appropriate choice for us, but we use them only after a certain level of discretization.

Pearlman and Smith (1998, 1999) describe a system of BCUT descriptors, which were derived from a method of Burden (1989). BCUTs are eigenvalues from connectivity matrices derived from the molecular graph. A connectivity matrix is obtained by placing an atomic property (e.g. size, atomic number, partial charge, etc.) for each heavy (non-hydrogen) atom along the diagonal of a square matrix. The off-diagonal elements of the matrix measure the degree of connectivity between two heavy atoms. The eigenvalues of this matrix then measure properties of the molecular graph, since they (eigenvalues) are matrix invariants. Furthermore, this construction allows the eigenvalues to be functions of all the heavy atoms

in the molecule. Following Lam (2001), we assume that they capture useful aspects of the properties of the molecule. Pearlman and Smith (1998, 1999) describe 67 BCUT numbers. Since these 67 BCUT descriptors are highly correlated, computational chemists typically use a subset of six fairly uncorrelated BCUT numbers. They tend to characterize molecular bonding patterns and atomic properties such as surface area, charge, hydrogen-bond donor, and acceptor ability. We will follow this approach.

Lam (2001), Stanton (1999) and Yi, Hughes-Oliver, Zhu, and Young (2002) all have shown utility of BCUTs as molecular descriptors. Having found molecular descriptors that capture useful properties and structural features of a compound, the question now arises of how these descriptors can be used to express and evaluate the two proposed design criteria of optimal coverage and minimal collision. We choose a data-driven binning approach, which follows a method proposed by LWY and will be described in detail in the next section.

5 DATA-DRIVEN BINNING AND SPACEFILL ALGORITHM

5.1 Data-driven Binning Method

One of our goals is to choose compounds that will be pooled in such a way that they cover the chemical space as completely as possible. To be able to measure whether the space is covered or not, we modify the BCUT descriptors. What we want is to divide the range of the BCUT descriptors into a rather fine grid, that will help us to obtain meaningful coverage of the chemical space. We should note that the original BCUT descriptors are continuous, but the objects (compounds) in the space are discrete; there will be areas in the chemical space that cannot be covered, since no objects exist.

[Figure 2 about here.]

11

Figure 2 shows histograms of the six BCUT descriptors. It can be seen that only a few compounds live in the outer ranges of the six descriptors. To assure that cells are better occupied, a hybrid method of combined equal width (EW) and equal frequency (EF) binning is used. Each descriptor range is divided into 64 cells. The cutoff values of the two outermost cells are chosen such that 1% of the compounds reside in each of these outer cells. The remaining inner bins are divided into 62 equal-width cells. Figure 3 illustrates this principle.

[Figure 3 about here.]

Our goal is to cover the chemical space. Therefore, it is not satisfactory to look only at the one-dimensional (1-D) subspaces spanned by the six BCUT descriptors. However, if we take all possible subspaces into account (up to 6-D subspaces), too many design points are needed to achieve a fine coverage of the chemical space, $(64)^6 = 6.9 \times 10^{10}$ cells. In HTS, activity is usually the result of multiple mechanisms that are observed through different descriptor sets and descriptor ranges (Rusinko, Farmen, Lambert, Brown, and Young 1999; Young and Hawkins 1998). Each activity mechanism is typically driven by a small subset of descriptors, and so it seems reasonable that these important descriptors will be captured by applying fine grids to low-dimensional subspaces of the 6 BCUT descriptors. We limit attention to all 1-D, 2-D, and 3-D subspaces spanned by the 6 BCUT descriptors, as suggested by LWY.

We now want to modify the proposed 1-D binning approach for the 2-D and 3-D subspaces. Originally, we choose 64 cells for the one-dimensional subspaces. Using the same grid for the 2-D subspace, we would obtain 4,096 cells, and 262,144 for the 3-D subspaces. Since we want to avoid this exponential increase of the number of cells for higher dimensions, we keep the same number of cells for each subspace. To form the cells for a 2-D subspace, we

divide each dimension into eight cells, $8 \times 8 = 64$. This is done by amalgamating eight cells of the 1-D subspace into one cell of the new 2-D subspace. This is done for all $\binom{6}{2} = 15$ possible 2-D subspaces for all six BCUT descriptors. In this way, each 2-D subspace is described again by 64 cells. Figure 4 illustrates this method.

[Figure 4 about here.]

Three-dimensional subspaces are formed by amalgamating 16 1-D bins into a single bin of the 3-D subspace. Each dimension is divided into 4 cells, so that the 3-D subspace is expressed by 64 cells, $4 \times 4 \times 4 = 64$. There are a total of $\binom{6}{3} = 20$ 3-D subspaces. In order to be able to measure the coverage of the chemical space and the number of collisions that occur in each pool, we define a new descriptor matrix. This new matrix contains all 20 3-D subspaces, each expressed by its 64 bins, which leads to a row vector of 1,280 elements for a specific molecule, or a 1,280-dimensional covariate space. The new descriptors are binary, and each descriptor indicates whether a compound resides in a certain cell of one particular 3-D subspace. When creating descriptors for pools, we combine the rows of the descriptor matrix for all the compounds grouped together in a particular pool. These pooled descriptors are again binary. If at least one compound in the pool has a one-entry in the original descriptor, the pooled descriptor is set to one as well. Figure 5 illustrates this concept and gives an example of the pooled descriptors of one 3-D subspace; the compound vectors are "ORed" to produce a bit string for the pool.

[Figure 5 about here.]

5.2 SpaceFill Algorithm

SpaceFill is a fast exchange algorithm that uniformly fills a space, as described by LWY. The following gives a brief description of the method. We then illustrate how we use the

13

algorithm to help fulfill our two design criteria.

The SpaceFill algorithm is based on covering all 1-D, 2-D, and 3-D subspaces as completely as possible. The goal of SpaceFill is to put a target number of compounds, usually one, into each cell occupied by a candidate molecule.

In subspace s, a measure of lack of uniformity is

$$U_s = \sum_i [n_{si}(X_d) - c_{si}(X_c)]^2, \tag{1}$$

where $n_{si}(X_d)$ is the number of design compounds in cell i of subspace s, $c_{si}(X_c) = 0$ if the candidate set X_c does not cover cell i of subspace s, and $c_{si}(X_c) = 1$ if the candidate set X_c covers cell i of subspace s.

To obtain the total lack of uniformity for all 3-D subspaces, the quantity U_s will be averaged over all three-dimensional subspaces:

$$U_{3-D} = \frac{\sum_{s \in S_3} U_s}{|S_3|} \tag{2}$$

Analogous definitions apply for U_{1-D} and U_{2-D}.

The uniform cell coverage criterion (UCC) of LWY is minimization of

$$UCC = \frac{U_{1-D} + U_{2-D} + U_{3-D}}{3}, \tag{3}$$

where the 1-D, 2-D, and 3-D subspaces can be individually weighted. UCC minimization can be achieved by having no uncovered candidate cells in the design, and placing only one design point, for example, per occupied cell.

Optimization in the SpaceFill algorithm is based on a basic exchange algorithm (Wynn 1972).

14

This algorithm randomly chooses n compounds from the set of all available N compounds. It then performs a series of exchanges, each attempting to improve UCC. An exchange is performed by first adding a design point from the candidate list to the evolving design of n compounds. The added compound gives an improved UCC value based on $n+1$ compounds. Next, one compound is removed from this evolving design by determining the smallest UCC value for a design with n compounds. This exchange process is repeated until no further reduction of the UCC value can be achieved.

The question left to be discussed is how do we use the SpaceFill algorithm to fulfill the two design criteria? Our first goal is to choose a subset of all the available compounds that fulfill the first design criterion, optimal coverage. We do this by applying the SpaceFill algorithm to our complete database and asking it to pick 4,096 design points, placed into 4,096 bins. The 4,096 bins refer to the total number of cells in each of the 1-D, 2-D, and 3-D subspaces. For the 1-D subspaces, this means a grid of 4,096 cells, in the 2-D subspaces 64×64 cells, and in the 3-D subspaces 16×16×16 cells. The SpaceFill algorithm aims to select the design points in such a way that each cell of all 1-D, 2-D, and 3-D subspaces is occupied by at least one compound. We choose a fine grid to guarantee good coverage of the entire chemical space.

The next step is to build the pools from these diverse compounds, addressing the second design criterion, minimal collision within a pool. We apply the SpaceFill algorithm to place compounds in pools. Compounds are chosen from among the 4,096 previously selected design points for designs OC, SF1, and SF6, and from the entire collection of compounds for design MC. This time we select 64 design points, placed in 64 bins. This refers to a grid of 64 cells in the 1-D subspaces, 8×8 cells in the 2-D subspaces, and 4×4×4 cells in the 3-D subspaces. SpaceFill will choose the points attempting to get one point in each cell of all the subspaces. To construct pools of size n, we will now randomly choose n compounds, out

of the 64 chosen design points. These, say 10, compounds should be rather different from one another. By selecting a much coarser grid than before (64 vs 4,096), we ensure that the selected compounds are far enough apart from each other to minimize collisions. Additional details on pool construction will be provided in the next section.

6 FOUR POOLING DESIGNS APPLIED TO NCI AIDS DATA

6.1 Design 1: Optimal Coverage

The first design, Optimal Coverage (OC), only tries to fulfill one design criterion, optimal coverage. As described in Section 5.2, the SpaceFill algorithm is used to choose 4,096 compounds that give the best possible coverage of the chemical space. Pools are created by randomly selecting 10 of these 4,096 compounds at a time, without replacement, and assigning them to the same pool. In this way, 409 pools are created (the remaining six compounds will be ignored). This design does not try to minimize collision; it is used as a benchmark design for assessing minimum collision. We created 20 replicates by repeatedly selecting 4,096 compounds using SpaceFill, and assigning them to pools at random.

6.2 Design 2: SpaceFill 1

SpaceFill 1 (SF1) aims to fulfill both design criteria. First, an initial set of 4,096 compounds is chosen in the same fashion as in the OC design. This step picks compounds that fulfill the criterion of optimal coverage. In order to be able to compare the two designs OC and SF1, we use for SF1 the same 4,096 compounds as in the OC design. The next step is building the pools, while fulfilling the design criterion of minimal collision. In Section 5.2, we gave a brief description of how we achieve this. SpaceFill is used to select a subset of 64 compounds among the previously chosen 4,096 compounds. By defining a much coarser grid, we assure that these compounds are not too close to each other in the chemical space. Now,

10 compounds are chosen at random and assigned to the first pool. Since all 64 selected compounds have a certain distance to each other, minimal collisions should occur within the 10 pooled compounds. To create the second pool, the above steps are repeated, using the subset of the remaining 4,086 compounds. We created 390 pools in this way. The SpaceFill algorithm was designed for large data sets and will not work with small candidate sets; in our case, we could not go below 300 candidate points. Again, 20 replicates were created.

6.3 Design 3: SpaceFill 6

The third design, SpaceFill 6 (SF6), is a slight modification of the SF1 design. The subset of 4,096 compounds and the first candidate set for the pools of 64 compounds are selected as described in Section 6.2. Again we use the same 4,096 compounds. We now randomly select 60 compounds among the 64 molecules, and randomize them into six pools of 10. To create pools 7 to 12, SpaceFill is used to select 64 compounds among the remaining 4036 compounds, and the above steps are repeated. We created 390 pools in this way. Twenty replicates were created. This method is more computationally efficient, but might not give as high a quality design.

6.4 Design 4: Minimized Collision

The fourth design, Minimized Collision (MC), is primarily aimed at fulfilling the design criterion of minimal collision. Without selecting a subset of compounds that is trying to cover the chemical space as completely as possible, we select compounds from the complete chemical library and assign them directly into pools. The compounds that will be assigned to the same pool should not be too close to each other in the chemical space to avoid collision. In particular, we use SpaceFill to select 64 compounds from all available 29,749 compounds. For each of the 1-D, 2-D, and 3-D subspaces, we have 64 cells. This number

of bins guarantees a coarse enough grid to choose 64 compounds that live far enough apart from each other to avoid collision. We then randomly select 10 compounds among the 64 compounds, and assign them to pool 1. SpaceFill is run again to select 64 compounds among the remaining 29,739 compounds. Pool 2 is created by again randomly selecting 10 compounds among the 64 selected compounds. This procedure is repeated until 390 pools are created. Twenty replicates were created.

6.5 Application to NCI AIDS Data

We illustrate our pooling design methods on the NCI AIDS Antiviral screen database of 29,749 compounds with 607 (2.04%) potent compounds. The chemical descriptors used are 6 BCUT numbers where each of the 20 3-D subspaces is binned into 64 bins giving a total of 1,280 binary covariates. This is described in more detail in Section 5.1. Results are given for five designs. All results are averaged across 20 replicates. Primary interest is in the four designs OC, SF1, SF6, and MC. The fifth design, TR, is used as a control/benchmark. Here, 390 pools are created by randomly selecting, without replacement, 10 compounds at a time from the complete NCI database.

Assessing Coverage.

Table 2 shows how well the optimal coverage criterion is fulfilled. The UCC criterion (see Section 5.2) is evaluated for compounds selected at the first step of the design. For each of the 20 replicates, we selected 4,096 compounds that cover the chemical space as completely as possible in all 1-D, 2-D, and 3-D subspaces. The UCC value was calculated for these compounds. OC, SF1 and SF6 share the same 4,096 compounds for each of the 20 replicates, so the Optimal Coverage line in Table 2 also applies to SF1 and SF6. In the MC design, we select compounds and assign them directly to pools using a coarse grid in

SpaceFill. However, to evaluate the UCC value, we use the 3,900 individual compounds of each replicate. Because a different number of compounds is evaluated here, it is more appropriate to compare the average percent of cells covered, pc, than comparing UCC. When calculating pc, we average the percentage of cells covered by a design over all subspaces. Cells that are not covered by the complete database, the candidate set X_c, are eliminated from consideration when pc is evaluated. Results are shown in Table 2.

[Table 2 about here.]

Table 2 shows that compounds selected with the SpaceFill algorithm in OC give significantly better coverage of the chemical space than the randomly selected compounds in TR (UCC: t = 178.56, p-value $\leq$ 0.0001; pc: t = 247.58, p-value $\leq$ 0.0001). The same holds for the MC design. It covers a higher percentage of cells averaged across all replicates than the TR design (UCC: t = 62.47, p-value $\leq$ 0.0001; pc: t = 140.17, p-value $\leq$ 0.0001).

Assessing Collision.

In a cell containing k compounds, $\binom{k}{2}$ collisions occur. To obtain the total number of collisions within a pool, we sum the number of collisions per cell over all cells corresponding to compounds in that pool. Figure 6 illustrates how well the five designs fulfill the criterion of minimal collision.

[Figure 6 about here.]

Table 3 shows collision averages across the 20 replicates, as well as standard deviations and minimum and maximum number of collisions for several different pool numbers. The pool number indicates the order in which the pools were created. In comparison to TR, all four designs (OC, SF1, SF6, and MC) perform much better with respect to collisions for the majority of the pools. SF1, SF6, and MC are the three methods that were designed with

19

the goal to minimize the number of collisions within each pool. All three designs perform better than OC for at least half of the pools created.

[Table 3 about here.]

SF1 and SF6 fulfill the design criterion equally well. There is no significant difference between the averaged number of collisions in SF1 and SF6 (t = 0.67, p-value = 0.5049). Figure 6 shows, for less than approximately 270 pools, SF1 and SF6 meet the design criterion better than OC. Because both methods choose only compounds from the preselected 4,096 compounds that cover the space, as the candidate set becomes smaller with an increasing pool number, the total number of collisions within the pools increases exponentially for SF1 and SF6. This increase suggests starting with a somewhat larger initial diverse set, say 7,000 rather than 4,096 compounds.

MC gives the best collision performance of all pooling designs. When selecting compounds that go into pools for the MC design, we are not as restricted in the choice of compounds since we choose from the complete chemical library. MC shows an excellent performance across all 390 pools created, and it significantly outperforms all other designs (t = 124.31, p-value ≤ 0.0001). This supports our conjecture that selecting 4,096 compounds as a first step, although computationally efficient, is too restrictive. The coverage step does, however, serve a useful function. The only difference between TR and OC in Figure 6 is the optimal coverage step. When performed, optimal coverage by itself significantly reduces collision (t = 82.16, p-value ≤ 0.0001).

Figure 7 shows the 10 compounds from a pool with low number of collisions (9 collisions) and Figure 8 shows the 10 compounds from a pool with high number of collisions (83 collisions). Figure 9 shows a comparison of the six BCUT values of the 10 compounds from the low collision pool, pool 1, and the 10 compounds from the high collision pool,

20

pool 390. Except for BCUT1, the descriptor values for pool 1 are more spread than the descriptor values for pool 390. The histogram of BCUT1 in Figure 2 shows that the values of the BCUT1 descriptor mainly fall in the area of -1. This explains why almost no difference can be seen in Figure 9 for the BCUT1 values of the 10 compounds from pool 1 and the 10 compounds from pool 390. Therefore, with respect to our descriptor space, the 10 compounds from the low collision pool are more diverse, or more structurally different than the 10 compounds from the high collision pool.

A visual comparison of compounds in pools 1 and 390 quickly reveals that compounds in pool 390 have more aromatic rings than compounds in pool 1. Molecular weights of the 10 compounds from the high collision pool range from 213 to 546, with most compounds being close to 400. Molecular weights of the 10 compounds from the low collision pool range from 134 to 525, showing more diversity than the high collision pool. However, although compounds in the high collision pool seem to be more similar to one another with respect to the number of aromatic rings and molecular weight than compounds in the low collision pool, this does not translate into chemical similarities with respect to specific chemical functional groups that correspond to similar binding behavior. Compounds in the high collision pool are still very diverse in terms of shape, rigidity and other properties that may be important in terms of binding abilities. In order to capture chemical features that are more significant for binding abilities, such as shape and rigidity, other chemical descriptors need to be investigated.

[Figure 7 about here.]

[Figure 8 about here.]

[Figure 9 about here.]

Synergism, Blocking and Additivity as Functions of Collision.

As mentioned earlier, OC, SF1, and SF6 each use the same set of 4,096 individual compounds for each replicate. Therefore, the differences we observe in the average number of collisions occurring in each pool are due entirely to the different methods of pooling these 4,096 individual compounds. Initially, SF1 and SF6 select compounds that have minimal collisions but after 270 pools are created, the number of collisions occurring in each pool increases exponentially. Both areas, low collision and high collision, can be of interest to the researcher.

If low collision occurs within a pool, the compounds in this pool live far away from one another in the chemical space. Residing in different areas of the chemical space is interpreted by us and others as being structurally dissimilar to one another. Compounds that are structurally different might have, when combined, a higher potential to act synergistically with each other or to block one another. In the area of combination therapy, it can be of interest to study synergism and blocking at an early stage in the drug discovery process. Pools with a low number of collisions, and therefore structurally different compounds, could provide an excellence strategy to study these effects. Blocking can also cause a high rate of missed positive compounds, called false negatives, since typically no retesting is done on individual compounds that reside in an inactive pool. Being aware of pools with a high potential of blocking compounds can help to decrease the number of false negatives; a mathematical model can point to potentially active compounds that reside in blocked pools.

Pools with a high number of collisions occurring in them contain compounds that live closer to each other in the chemical space. If compounds are very similar to each other, they have a high potential for additivity. Additivity is the effect of combining weakly active similar individual compounds to get significant activity when tested in a pool. This is similar to testing the compounds at a higher concentration. Typically, we are not interested

in compounds that are only active when tested at high concentrations. When looking for a new lead compound, high potency compounds are desired for several reasons. First, a smaller amount of the highly active compound is needed to get activity. Second, a highly active compound is more likely to be selective as it more exactly fits the target protein or receptor, and by inference it is less likely to fit another protein well, hence the compound is expected to be less toxic. The TR design has high number of collisions, which is not desirable.

SIMILARITY OF DESIGNS ACROSS REPLICATES AND DESIGN CRITERIA.

We first consider the similarity of compounds across the 20 replicates. We look at compounds that are chosen at the first stage of the design, when trying to obtain optimal coverage. As OC, SF1, and SF6 are based on the same sets of 4,096 compounds, we only give results for OC. Results for MC are based on the 3,900 individual compounds that form the pools. These results are summarized in Table 4. The notation $sim@m$ indicates the number of compounds simultaneously appearing in m of the 20 replicates. Table 4 shows that in the OC design a majority of the compounds are included in all 20 replicates. This explains the small variance of number of collisions for OC, SF1 and, SF6 in comparison to TR, as seen in Table 3.

[Table 4 about here.]

Further investigation of the 2,105 repeatedly included compounds for OC show that these compounds come from areas of the space where not too many compounds are available to simultaneously obtain good coverage for all 1-D, 2-D,and 3-D subspaces. For example, in the 1-D subspace for BCUT1, 131 of the 2,105 compounds are the only compounds existing in their cells. Hence, in order to cover these cells, the design must always select the same compounds. Averaged across all six 1-D subspaces, there are 178 cells on average in each

23

1-D subspace that only contain compounds from the 2,105 repeatedly included compounds. The SpaceFill algorithm selects these compounds over and over because it has no other compounds to choose from these cells. Since the UCC value for OC is significantly better than the UCC value for TR, we can assume that these 2,105 repeatedly selected compounds play an important role in achieving a good coverage of the chemical space.

7   SUMMARY AND CONCLUSIONS

We constructed four pooling designs that fulfill one or both of the proposed design criteria of optimal coverage and minimal collision. All four designs provide better coverage of the chemical space than the benchmark design, TR. SF1, SF6, and MC also create pools with a very low number of colliding compounds. The collision performance of SF1 and SF6 decreases exponentially after a certain number of pools have been created, thus producing both low-collision and high-collision pools. SF6 is preferred over SF1 on the basis of computational efficiency, since SF6 creates 6 pools for every 1 pool created by SF1. MC uniformly achieves fewer collisions than all other designs for all pools created. The drawback of MC is that it is not as computationally efficient as SF1 and SF6 since, for each step in the exchange algorithm, a much larger candidate set of compounds must be searched. Additionally, UCC is not as small as in OC, SF1, and SF6.

Summarizing, we can say that all four pooling designs achieve dramatically better performance when compared to randomly constructed pools for both optimal coverage and minimal collision. They offer an opportunity to screen compounds in a more efficient but also more informative way. These designs open the possibility to study the effects of blocking and synergism. Among all designs considered here, MC has the best performance.

Formal analysis of pooled screening data obtained from these pooling designs is planned for future research. Zhu et al. (2001) and Yi (2002) suggest model-based likelihood ap-

24

proaches for the analysis of pooling designs. Both include the effects of blocking and the latter includes the effect of synergism in their models. These analysis techniques, as well as the more standard recursive partitioning (Young and Hawkins 1998; Rusinko et al. 1999), will be applied to pooled data obtained through our proposed pooling designs.

REFERENCES

ActivBiotics (2003), "Corporate Fact Sheet, January 2003," `http://www.activbiotics.com/pdfs/activbiotics_fac_sheet.pdf`.

Behets, F., Bertozzi, S., Kasali, M., Kashamuka, M., Atikala, L., Brown, C., Ryder, R. W., and Quinn, T. C. (1990), "Successful Use of Pooled Sera to Determine HIV-1 Seroprevalence in Zaire With Development of Cost-Efficiency Models," *AIDS*, 4, 737-741.

Burden, F. R. (1989), "Molecular Identification Number for Substructure Searches," *Journal of Chemical Information and Computer Sciences*, 29, 225-227.

Carhart, R. E., Smith, D. H., and Ventkataraghavan, R. (1985), "Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Application," *Journal of Chemical Information and Computer Sciences*, 25, 64-73.

Cummins, D. J., Andrews, C. W., Bentley, J. A., and Cory, M. (1996), "Molecular Diversity in Chemical Databases of Commercially Available Compounds," *Journal of Chemical Information and Computer Sciences*, 36, 750-763.

Dorfman, R. (1943), "The Detection of Defective Members of Large Populations," *Annals of Mathematical Science*, 14, 436-440.

Hann, M., Hudson, B., Lewell, X., Lifely, R., Miller, L., and Ramsden, N. (1999), "Strategic

Pooling of Compounds for High-Throughput Screening," *Journal of Chemical Information and Computer Sciences*, 39, 897-902.

Hughes-Oliver, J. M., and Rosenberger, W. F. (2000), "Efficient Estimation of the Prevalence of Multiple Traits," *Biometrika*, 87, 315-327.

Lam, R. L. H. (2001), "Design and Analysis of Large Chemical Databases for Drug Discovery," unpublished Ph.D. dissertation, University of Waterloo, Dept. of Statistics.

Lam, R. L. H., Welch W. J., and Young, S. S. (2002), "Uniform Coverage Designs for Molecule Selection," *Technometrics*, 44, 99-109.

Langfeldt, S. A., Hughes-Oliver, J. M., Ghosh, S., and Young, S. S. (1997), "Optimal Group Testing in the Presence of Blockers," Institute of Statistics Mimeograph Series 2297, North Carolina State University, Dept. of Statistics.

McFarland, J. W., and Gans, D. J. (1986), "On the Significance of Clusters in the Graphical Display of Structure-Activity Data," *Journal of Medicinal Chemistry*, 29, 505-514.

Menard, P. R., Mason, J. S., Morize, I., and Bauerschmidt, S. (1998), "Chemistry Space Metrics in Diversity Analysis, Library Design, and Compound Selection," *Journal of Chemical Information and Computer Sciences*, 38, 1204-1213.

Nilakantan, R., Bauman. N., Dixon, J. S., and Ventkataraghavan, R. (1987), "Topological Torsion: A New Molecular Descriptor for SAR Applications. Comparison With Other Descriptors," *Journal of Chemical Information and Computer Sciences*, 27, 82-85.

Pearlman, R. S., and Smith K. M. (1998), "Novel Software Tools for Chemical Diversity," *Perspectives in Drug Discovery and Design*, 9/10/11, 339-353.

— (1999), "Metric Validation and the Receptor-Relevant Subspace Concept," *Journal of Chemical Information and Computer Sciences*, 39, 28-35.

Phatarfod, R. M., and Sudbury, A. (1994), "The Use of a Square Array Scheme in Blood Testing," *Statistics in Medicine*, 13, 2337-2343.

Rossow, K. L., and Janknecht, R. (2003), "Synergism Between P68 RNA Helicase and the Transcriptional Coactivators CBP and P300," *Oncogene*, 22, 151-156.

Rusinko, A., Farmen, M. W., Lambert C. G., Brown, P. L., and Young, S. S. (1999), "Statistical Analysis of a Large Structure/Biological Activity Data Set Using Recursive Partitioning," *Journal of Chemical Information and Computer Sciences*, 38, 1017-1026.

Stanton, D. T. (1999), "Evaluation and Use of BCUT Descriptors in QSAR and QSPR Studies," *Journal of Chemical Information and Computer Sciences*, 39(1), 11-20.

Tipping, A., Mahon, F. X., Zafirides, G., Lagarde, V., Goldman, J. M., and Melo, J. V. (2002), "Drug Responses of Imatinib Mesylate-Resistant Cells: Synergism of Imatinib With Other Chemotherapeutic Drugs," *Leukemia*, 16, 2349-2357.

Tu, X. M., Litvak, E., and Pagano, M. (1995), "On the Information and Accuracy of Pooled Testing in Estimating Prevalence of a Rare Disease: Application to HIV Screening," *Biometrika*, 82, 287-297.

Wynn, H. P. (1972), "Results in the Theory and Construction of D-Optimum Experimental Designs," *Journal of the Royal Statistical Society*, Ser. B, 34, 133-147.

Xie, M., Tatsuoka, K., Sacks, J., and Young, S. S. (2001), "Group Testing With Blockers and Synergism," *Journal of the American Statistical Association*, 96, 92-102.

Yi, B. (2002), "Nonparametric, Parametric and Semiparametric Models for Screening and Decoding Pools of Chemical Compounds," unpublished Ph.D. dissertation, North Carolina State University, Dept. of Statistics.

Yi, B., Hughes-Oliver, J. M., Zhu, L., and Young, S. S. (2002), "A Factorial Design to Optimize Cell-Based Drug Discovery Analysis," *Journal of Chemical Information and Computer Sciences*, 42, 1221-1229.

Young, S. S., and Hawkins, D. M. (1998), "Using Recursive Partitioning to Analyze a Large SAR Data Set," *Structure-Activity Relationship and Quantitative Structure-Activity Relationship*, 8, 183-193.

Zhu, L. (2000), "Statistical Decoding and Designing of Pooling Experiments Based on Chemical Structure," unpublished Ph.D. dissertation, North Carolina State University, Dept. of Statistics.

Zhu, L., Hughes-Oliver, J. M., and Young, S. S. (2001), "Statistical Decoding of Potent Pools Based on Chemical Structure," *Biometrics*, 57, 922-930.

List of Figures

Figure 1: One-way Pooling experiment where pooling is by column. Pool 1 illustrates synergism and Pool 8 illustrates blocking. Pools 4 and 11 show regular activity.
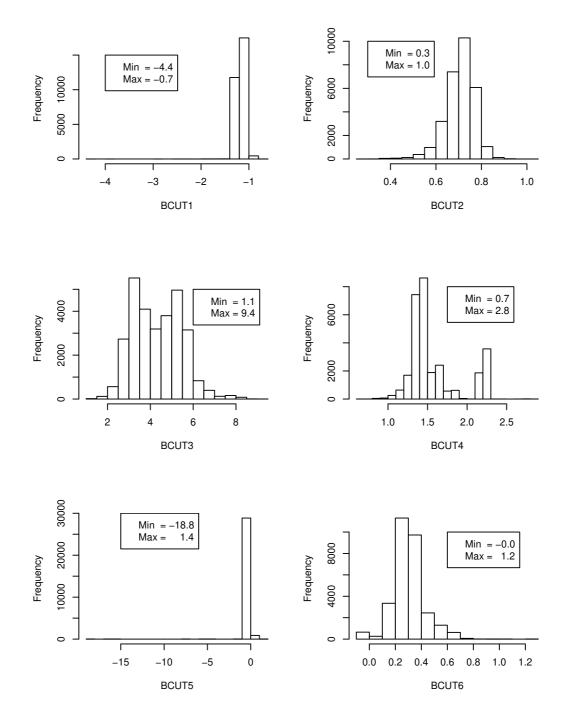
Figure 2: Histograms of six BCUT descriptors for the NCI data.
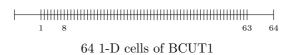
64 1-D cells of BCUT1

Figure 3: Equal Frequency and Equal Width Binning. Outer cells are formed by placing 1% of the compounds into them (EF). Remaining 62 cells have equal width.
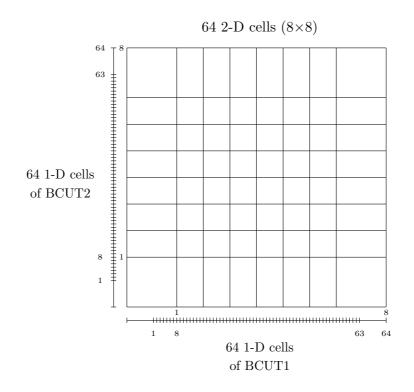
64 2-D cells (8×8)

64 1-D cells
of BCUT2

64 1-D cells
of BCUT1

Figure 4: Two-dimensional subspace. The 64 cells of the 2-D subspace are formed by amalgamating eight 1-D bins into one 2-D bin.

	Cell$_1$	Cell$_2$	Cell$_3$		Cell$_{63}$	Cell$_{64}$
Compound$_1$	0	0	1		1	0
Compound$_2$	0	1	0		1	0
Compound$_3$	0	1	0		1	0
Compound$_4$	0	0	0		0	0
Compound$_9$	0	0	0		0	0
Compound$_{10}$	0	1	0		0	1
Pool	0	1	1		1	1

Figure 5: Example of pool descriptors for one 3-D subspace. We "or" the bits; the pool descriptor is 1 if any compound in the pool is 1; otherwise the pool descriptor is 0.
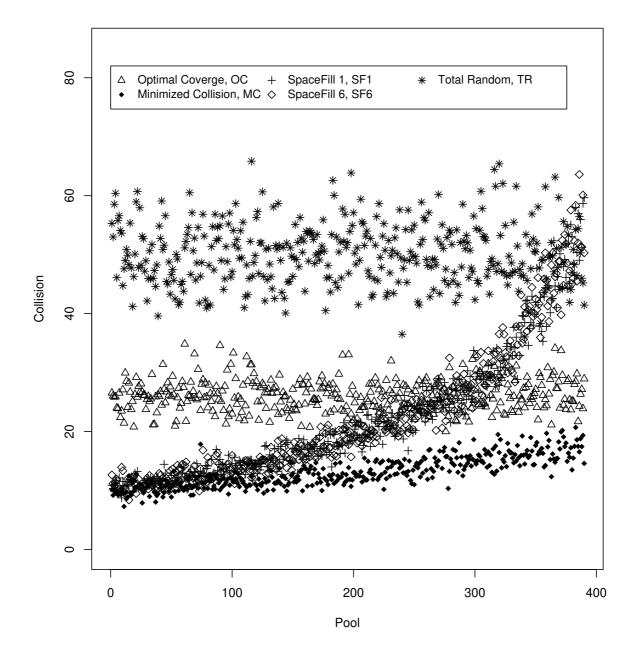
Figure 6: Collision plot. Twenty replicates were created for each method. Number of collisions is averaged across all 20 replicates. Pools are plotted in the order they were built.
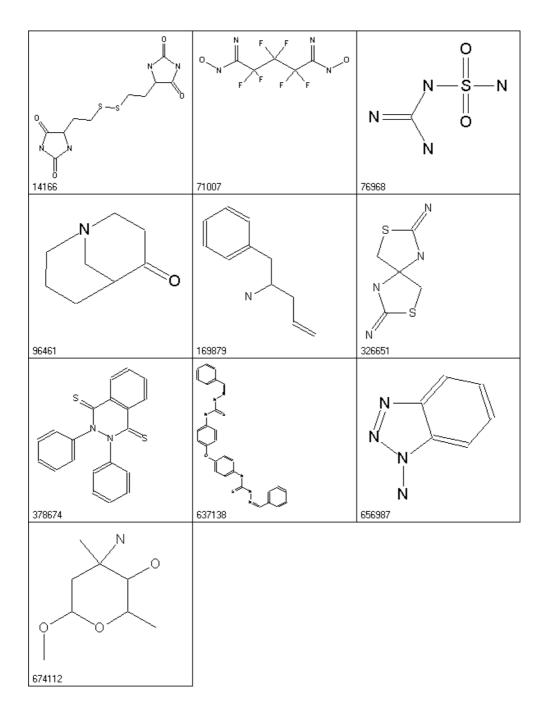
Figure 7: Compounds from a low collision pool. Only 9 collisions occur among these 10 compounds.
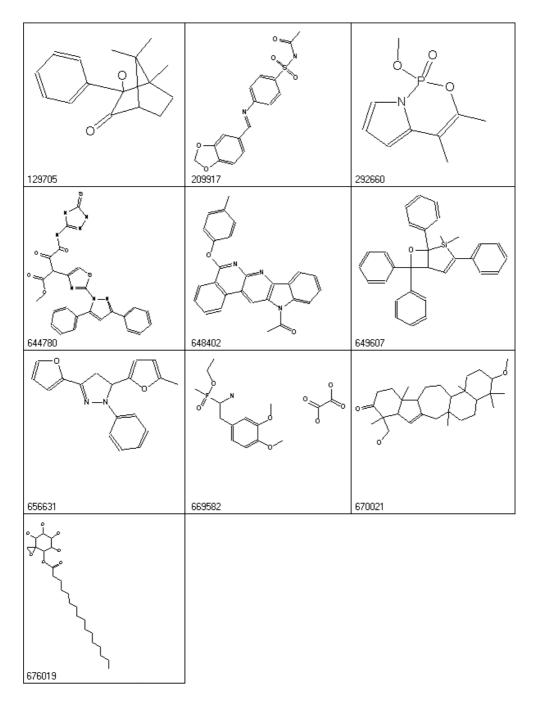
Figure 8: Compounds from a high collision pool, with 83 collisions occurring among the 10 compounds. All compounds have similar molecular weight and contain more aromatic rings.
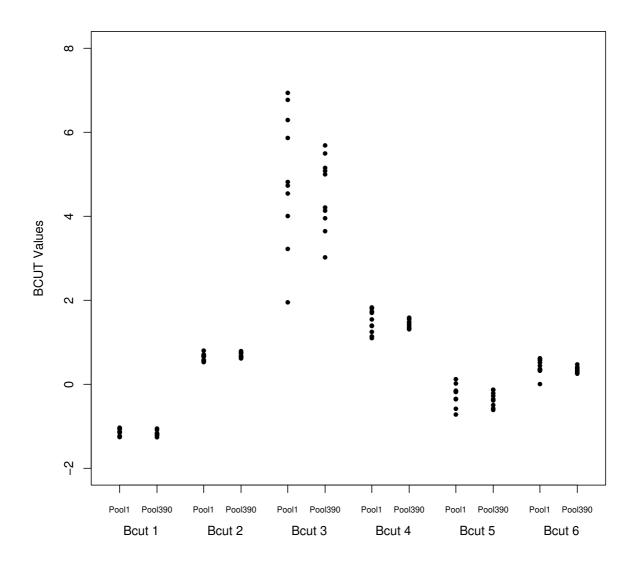
Figure 9: BCUT values for the 10 compounds in a low collision pool, pool 1, and for the 10 compounds in a high collision pool, pool 390. The BCUT values for the compounds in pool 1 are more spread than the values for the compounds in pool 390.

38

Table 1: *Four pooling designs (OC, MC, SF1, and SF6) and a benchmark pooling design, Total Random (TR). The table illustrates which design criteria are satisfied (has value 1) by each pooling design.*

	Design Criteria	
Design	Optimal Coverage	Minimal Collision
Total Random, TR	0	0
Optimal Coverage, OC	1	0
Minimized Collision, MC	0	1
SpaceFill 1, SF1	1	1
SpaceFill 6, SF6	1	1

Table 2: *UCC and percent of cells covered, pc, averaged across 20 replicates. UCC and pc are evaluated for compounds chosen in the first stage of the design. OC, SF1 and SF6 use the same 4,096 compounds for each of the 20 replicates. UCC and pc for the MC design is based on 3,900 compounds. Small UCC value implies better coverage of the candidate set by the selected subset.*

	UCC value			
Design	Mean	SD	Min	Max
Total Random, TR	22355.1	415.9	21572.3	22991.1
Minimized Collision, MC	14647.2	362.6	13916.3	15275.8
Optimal Coverage, OC	5745.9	6.9	5731.7	5757.7

	Percent Covered (pc)			
Design	Mean	SD	Min	Max
Total Random, TR	55.5	0.42	54.6	56.2
Minimized Collision, MC	70.2	0.21	70.0	70.7
Optimal Coverage, OC	79.1	0.06	79.0	79.3

Table 3: *Number of collisions across all pools, for pool 1, pool 270, and pool 390, averaged across all 20 replicates; Mean, Standard Deviation, Minimum, and Maximum.*

	OC				MC			
Collision	All	Pool 1	Pool 270	Pool 390	All	Pool 1	Pool 270	Pool 390
Mean	26.1	26.6	28.3	29.0	13.2	10.2	15.1	14.6
SD	11.9	7.9	12.9	7.9	6.7	3.2	5.3	5.7
Min	3	15	7	13	1	5	7	8
Max	108	44	55	48	191	17	24	26

	SF1				SF6			
Collision	All	Pool 1	Pool 270	Pool 390	All	Pool 1	Pool 270	Pool 390
Mean	22.3	10.9	29.0	59.7	22.5	12.7	24.4	50.3
SD	14.5	2.9	12.2	21.6	14.7	5.5	9.7	15.5
Min	0	5	9	30	1	5	4	26
Max	161	16	55	106	198	25	45	83

	TR			
Collision	All	Pool 1	Pool 270	Pool 390
Mean	50.2	55.3	60.2	41.5
SD	22.9	27.5	29.9	13.4
Min	8	20	23	21
Max	219	104	143	62

Table 4: *Similarity of compounds across 20 replicates. sim@m is the number of compounds simultaneously appearing in m of the 20 replicates.*

m	20	19	18	17	16	15	14	13	12	11	10
TR, $sim@m$	0	0	0	0	0	0	0	0	0	0	7
OC, $sim@m$	2105	279	245	184	193	161	154	150	167	162	166
MC, $sim@m$	1399	322	203	183	183	175	168	162	195	185	201

m	9	8	7	6	5	4	3	2	1	0
TR, sim@m	15	90	319	1011	2541	4749	7055	7435	4960	1567
OC, sim@m	169	208	206	230	266	300	415	569	1006	22414
MC, sim@m	247	277	310	351	437	561	775	1199	2874	19342